Regular Correspondence:
195 North Harbor Drive, Suite 5403, Chicago Illinois 60601-7540

Docketed Correspondence:
Post Office Box 7131, Chicago Illinois 60680-7131

Peter K. Trzyna, Esq.

Telephone: (312) 240-0824   Facsimile: (312) 240-0825

E-mail: pktlaw@email.msn.com

RECEIVED
CENTRAL FAX CENTER

JAN 2 6 2005

Fax

To:	Examiner Patrice Winder	**Re:** 09/399,578 IDS and Cited Art	
Firm:	United States Patent and Trademark Office	**Date / Time:** January 26, 2005	
Street Address:		**Phone:** (571) 272-3935	
City, State Zip:	Washington, D.C., 20231	**Fax:** (703) 872-9306	
cc:		**No. of Pages:** 22 (including cover)	

Message:

```
                                        ksfo-client.txt
#!/usr/local/bin/perl

# WebChat(tm) Client v 0.2
# Copyright (c) 1995 Internet Roundtable Society
# Programmed by Michael Fremont, email: webchat@irsociety.com

#     This program is free software; you can redistribute it and/or modify
#     it under the terms of the GNU General Public License as published by
#     the Free Software Foundation; either version 2 of the License, or
#     (at your option) any later version.

#     This program is distributed in the hope that it will be useful,
#     but WITHOUT ANY WARRANTY; without even the implied warranty of
#     MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#     GNU General Public License for more details.

#     You should have received a copy of the GNU General Public License
#     along with this program; if not, write to the Free Software
#     Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.


# this script is executed when a user submits the "chat" form.

#LITERALS
#--------
$LOCK_SH = 1;
$LOCK_EX = 2;
$LOCK_NB = 4;
$LOCK_UN = 8;

$TRUE    = 1;
$FALSE   = 0;

#GLOBALS
#-------
$client                                 = "http:/cgi-bin/nph-client#anchor1";

# the following line is for the www.irsociety.com system
#$talkfile                              = "/home/webchat/transcripts/ksfo";
# the following line is for the webchat.service.digital system
#$talkfile                  =
"/usr/local/httpd/htdocs/webchat/transcripts/ksfo";
$talkfile                  =
"/home/lanshark/www/pages/webchat/transcripts/ksfo";

$point_gif                 =
"http://www.cybertoday.com/cybertoday/webchat/point.gif";
$webchat_logo              =
"http://www.ccnet.com/laporte/images/ksfologo.gif";
$about_local_server                     = "http://www.cybertoday.com/";
$local_tz                               = "PDT (-0700 GMT)";

$read_block_size                        = 512;
$num_context_paras                      = 0;
$num_context_paras_when_starting        = 10;
$para_mark_size                         = 7;    # 7 digits (space padded)
$back_jump                              = 10;
$way_back_when                          = 40;

# get the form
&ReadParse;
```

Page 1

```
                                  ksfo-client.txt
$last_read_para = $in{last_read_para};
$wants_dates_printed = $in{wants_dates_printed};
$back_para = $in{back_para};

# we've changed to letting you specify how far back you want to scroll
# we're leaving the old code in case we go back
if($in{Back} > 0)
        {
        $back = $TRUE;
        $back_para = $back_para - $back_jump;
        ($back_para < 1) && ($back_para = 1);
#       $last_read_para = $back_para;
        $last_read_para = $last_read_para - $in{Back};
        ($last_read_para < 1) && ($last_read_para = 1);
        }




#!!! for debugging
#if($in{InputText} eq "")
#       {
#       $in{handle} = "Michael Fremont";
#       $in{InputText} = "hello, world!";
#       }

# open a (properly initialized) transcript file

# later: if need to die, put text in some log file somewhere
open (TRANSCRIPT, "+<$talkfile") || die "client
can't open transcript file";

# If the user input any text, add it to transcript file
($in{InputText} ne "" && $back eq "") && &add_to_transcript;

# Update the output area or send error message if nothing new
&output_new_form;
exit;


sub output_new_form

# if there is new output for this user, send it to him.  Otherwise return
# "no new info" error message to his browser so it keeps his current state

{
local ($buf, $tbuf, $last_para, $found, $amount_to_read);
local ($para_num, $date, $handle, $headURL, $headURLsize, $text);



# look for first context paragraph that is to be output to user
# note - it may not be in transcript file anymore, if the file was pruned
# but for this version, assume pruning has not yet been implemented

$first_context_para = $last_read_para - $num_context_paras;
($first_context_para < 1) && ($first_context_para = 1);
$ffirst_context_para =
        "\0" .
        pack("A$para_mark_size", $first_context_para) .
        "\1";

# read a block from the end of the file and look for the context para.
                                Page 2
```